

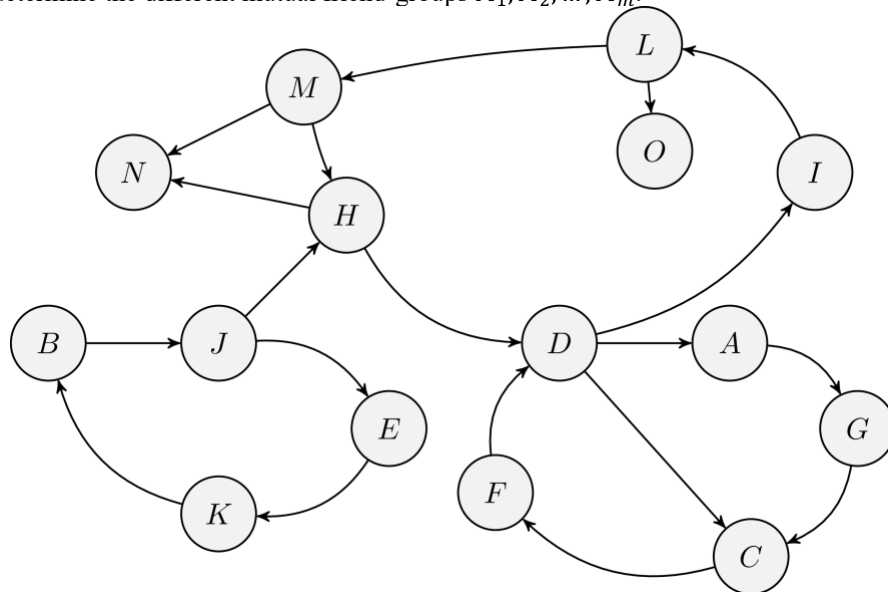
## 11. Wang

**Program Name:** Wang.java

**Input File:** wang.dat

Wang is quite the social butterfly and loves getting to know new people on the regular. However, due to the breadth of her friendships with so many people, Wang often gets confused with which of her friends are mutually friends with one another. This makes it hard to know what friends to invite to different functions, since she does not want to force others to socialize with people they are not comfortable with.

As far as Wang is concerned, she considers friendship to be a directional relationship, rather than a unidirectional (or bidirectional) relationship. That is, person  $A$  may feel as though person  $B$  is their friend, but person  $B$  may not inherently be too fond of person  $A$ . Moreover, Wang will consider the definition of a “mutual friend-group” to be the following: let  $P$  be the set of all people. The set  $M$  is a mutual friend-group ( $M \subseteq P$ ) so long as for every person  $p$  in  $P$ , they are either directly or indirectly friends with every other person in  $M$  (denoted  $\forall q \in M, p \rightsquigarrow q$ ). More formally, for every pair of individuals  $p$  and  $q$  in  $M$ , there exists some path from  $p$  to  $q$  in the directed graph (we will assume that every person inherently thinks of themselves as a friend). Note that Wang will only consider a set of mutual friends to be a mutual friend-group so long as there are no other people in  $P$  that could possibly belong to the friend group. Given a set of people  $P$ , and the set of their perceived friendship relations, help Wang determine the different mutual friend-groups  $M_1, M_2, \dots, M_m$ .



**Input:** First line of input will consist of a single integer,  $T$  ( $1 \leq T \leq 500$ ), denoting the number of test cases to follow. Each test case will begin with two single space-separated integers  $n_i$  ( $2 \leq n_i \leq 10^4$ ) and  $e_i$  ( $1 \leq e_i \leq 2 \cdot 10^4$ ), denoting the number of people in  $P$  and the number of perceived friendship relations for the  $i^{\text{th}}$  test case. The next line will consist of  $n_i$  single space-separated strings denoting the names of each of the  $n_i$  people in  $P$ . You may assume that all names are unique and only comprise of alphanumeric characters. The following  $e_i$  lines will consist of two single space-separated strings, the  $j^{\text{th}}$  of which denotes the names of the two people involved in the  $j^{\text{th}}$  friendship relation. You may assume that the first name that appears believes that they are friends with the second name that appears on each of the  $e_i$  lines. Lastly, you may assume that the only names that will appear on the  $e_i$  lines will be names that appear among the  $n_i$  names of  $P$ .

~ Wang continues next page ~

~ Wang *continued* ~

**Output:** For each of Wang's  $T$  queries, on its own line, print out the set of mutual friend-groups. Each mutual friend group will take the format of " $\{p_1, p_2, \dots, p_\ell\}$ ", where  $p_1$  is the lexicographically smallest name among the  $\ell$  members of the group,  $p_2$  is the second smallest, and so on. Mutual friend groups should be separated by a single comma followed by a single space, and should be listed first by decreasing set size, and then by lexicographical order in the case of ties.

**Sample input:**

```
1
15 19
A B C D E F G H I J K L M N O
B J
J E
E K
K B
D I
I L
L M
M H
H D
A G
G C
C F
F D
D A
J H
M N
L O
D C
H N
```

**Sample output:**

```
{A, C, D, F, G, H, I, L, M}, {B, E, J, K}, {N}, {O}
```